Selective Attention for Context-aware Neural Machine Translation

Sameen Maruf[†][*] **André F. T. Martins**[‡] **Gholamreza Haffari**[†]
[†]Faculty of Information Technology, Monash University, VIC, Australia
[‡]Unbabel & Instituto de Telecomunicações, Lisbon, Portugal
[†]{firstname.lastname}@monash.edu
[‡]andre.martins@unbabel.com

Abstract

Despite the progress made in sentence-level NMT, current systems still fall short at achieving fluent, good quality translation for a full document. Recent works in context-aware NMT consider only a few previous sentences as context and may not scale to entire documents. To this end, we propose a novel and scalable top-down approach to hierarchical attention for context-aware NMT which uses sparse attention to selectively focus on relevant sentences in the document context and then attends to key words in those sentences. We also propose single-level attention approaches based on sentence or word-level information in the context. The document-level context representation, produced from these attention modules, is integrated into the encoder or decoder of the Transformer model depending on whether we use monolingual or bilingual context. Our experiments and evaluation on English-German datasets in different document MT settings show that our selective attention approach not only significantly outperforms context-agnostic baselines but also surpasses context-aware baselines in most cases.

1 Introduction

Neural machine translation has grown immensely in the past few years, from the simplistic RNN-based encoder-decoder models (Sutskever et al., 2014; Bahdanau et al., 2015) to the state-of-the-art Transformer architecture (Vaswani et al., 2017). Most of these models rely on the attention mechanism as a major component, which involves focusing on different parts of a sequence to compute new representations, and has proven to be quite effective in improving the translation quality (Vaswani et al., 2017). However, all of these models share the same inherent problem: the translation is still performed on a sentence-by-sentence basis, thus ignoring the long-range dependencies which may be useful when it comes to translating discourse phenomena.

More recently, **context-aware NMT** has been gaining significant traction from the MT community with majority of works coming out in the past two years. Most of these focus on using a few previous sentences as context (Jean et al., 2017; Wang et al., 2017; Tu et al., 2018; Voita et al., 2018; Zhang et al., 2018; Miculicich et al., 2018) and neglect the rest of the document. Only one existing work has endeavoured to consider the full document context (Maruf and Haffari, 2018), thus proposing a more generalised approach to document-level NMT. However, the model is restrictive as the document-level attention computed is sentence-based and static (computed only once for the sentence being translated). A more recent work (Miculicich et al., 2018) proposes to use a hierarchical attention network (HAN) (Yang et al., 2016) to model the contextual information in a structured manner using word-level and sentence-level abstractions; yet, it uses a limited number of past source and target sentences as context and is not scalable to entire document.

In this work, we propose a **selective attention** approach to first selectively focus on relevant sentences in the global document-context and then attend to key words in those sentences, while ignoring the rest.[1] Towards this goal, we use **sparse attention**, enabling an efficient and scalable use of the context. The intuition behind this is the way humans translate a sentence containing ambiguous words. They may look for sentences in the whole document which contain similar words and just focus on those for the translation. This attention,

[*] Work initiated during an internship at Unbabel.

[1] The term "selective attention" comes from cognitive science and is defined as the act of focusing on a particular object for a period of time while simultaneously ignoring irrelevant information that is also occurring (Dayan et al., 2000).

Proceedings of NAACL-HLT 2019, pages 3092–3102
Minneapolis, Minnesota, June 2 - June 7, 2019. ©2019 Association for Computational Linguistics

which we call Hierarchical Attention, is computed dynamically for each query word. Furthermore, we propose a Flat Attention approach which is based on either sentence or word-level information in the context. We integrate the document-level context representation, produced from these attention modules, into the encoder or decoder of the Transformer model depending on whether we consider monolingual (source-side) or bilingual (both source and target-side) context.

Our contributions are as follows: (i) we propose a novel and efficient top-down approach to hierarchical attention for context-aware NMT, (ii) we compare variants of selective attention with both context-agnostic and context-aware baselines, and (iii) we run experiments in both online (only past context) and offline (both past and future context) settings on three English-German datasets. Experiments show that our approach improves upon the Transformer by an overall +1.34, +2.06 and +1.18 BLEU for TED Talks, News-Commentary and Europarl, respectively. It also outperforms two recent context-aware baselines (Zhang et al., 2018; Miculicich et al., 2018) in majority of the cases.

2 Background

2.1 Neural Machine Translation

Generic NMT models are based on an encoder-decoder architecture (Bahdanau et al., 2015; Vaswani et al., 2017). The encoder reads the source sentence denoted by $\boldsymbol{x} = (x_1, x_2, ..., x_M)$ and maps it to a continuous representation $\boldsymbol{z} = (z_1, z_2, ..., z_M)$. Given $\boldsymbol{z}$, an attentional decoder generates the target translation $\boldsymbol{y} = (y_1, y_2, ..., y_N)$ one word at a time in a left-to-right fashion. The popular Transformer architecture (Vaswani et al., 2017) follows the same structure by using stacked self-attention and point-wise, fully connected layers for both the encoder and decoder.

Encoder The encoder stack is composed of L identical layers, each containing two sub-layers. The first, a multi-head self-attention sub-layer, allows each position in the encoder to attend to all positions in the previous layer of the encoder, while the second, a feed-forward network, uses two linear transformations with a ReLU activation.

Decoder The decoder stack is also composed of L identical layers. In addition to the two sub-layers, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Masking is used in the self-attention sub-layer to prevent positions from attending to subsequent positions thus avoiding left-ward flow of information.

2.2 Document-level Machine Translation

In general, the probability of a document translation $\boldsymbol{Y}$ given the source document $\boldsymbol{X}$ is given by:

$$P_{\boldsymbol{\theta}}(\boldsymbol{Y}|\boldsymbol{X}) = \prod_{j=1}^{J} P_{\boldsymbol{\theta}}(\boldsymbol{y}^j|\boldsymbol{x}^j, \boldsymbol{D}_{-j}) \qquad (1)$$

where $\boldsymbol{y}^j$ and $\boldsymbol{x}^j$ denote the j^{th} target and source sentence, respectively, and $\boldsymbol{D}_{-j} = \{\boldsymbol{X}_{-j}, \boldsymbol{Y}_{-j}\}$ is the collection of all other sentences in the source and target document. Since generic NMT models translate one word at a time, Eq. 1 becomes:

$$P_{\boldsymbol{\theta}}(\boldsymbol{Y}|\boldsymbol{X}) = \prod_{j=1}^{J} \prod_{n=1}^{N} P_{\boldsymbol{\theta}}(y_n^j|\boldsymbol{y}_{<n}^j, \boldsymbol{x}^j, \boldsymbol{D}_{-j}) \qquad (2)$$

where y_n^j is the n^{th} word of the j^{th} target sentence and $\boldsymbol{y}_{<n}^j$ are the previously generated words.

Training The document-conditioned NMT model $P_{\boldsymbol{\theta}}(\boldsymbol{y}^j|\boldsymbol{x}^j, \boldsymbol{D}_{-j})$ is realised using a neural architecture and usually trained via a two-step procedure (Maruf and Haffari, 2018; Miculicich et al., 2018). The first step involves pre-training a standard sentence-level NMT model, and the second step involves optimising the parameters of the whole model, i.e., both the document-level and the sentence-level parameters.

Decoding To generate the best translation for a full document according to the document MT model, the problem of maximizing Eq. 1 is solved using a two-pass Iterative Decoding strategy (Maruf and Haffari, 2018): first, the translation of each sentence is initialised using the sentence-based NMT model; then, each translation is updated using the context-aware NMT model fixing the other sentences' translations.

3 Proposed Approach

The main goal of this paper is to have a document-level NMT model which is memory-efficient, scalable, and capable of listening to the entire document. To achieve this, we augment a sentence-level NMT model (the Transformer (Vaswani et al., 2017)) with an efficient hierarchical attention mechanism which has the ability to identify the key sentences in the document context and then attend to the key words within those sentences. As mentioned previously, we want to maximise $P_{\boldsymbol{\theta}}(\boldsymbol{y}^j|\boldsymbol{x}^j, \boldsymbol{D}_{-j})$, where we take $\boldsymbol{D}_{-j}$ to be either the monolingual source or bilingual source and target-side context in two settings: *offline*—the context comes from both past and future, and *online*—the context comes from only the past.

In this section, we show how to represent the document-level context using our Context Layer, how to regulate the information at the sentence and document-level using context gating and finally we present our integrated model.

3.1 Document-level Context Layer

The context $\boldsymbol{D}_{-j}$ is modeled via a single Document-level Context Layer comprising of two sub-layers: (i) a Multi-Head Context Attention sub-layer, and (ii) a Feed-Forward sub-layer, where the former consists of either a top-down Hierarchical Attention module or a Flat Attention module (explained shortly), and the latter is similar to the Feed-Forward network in the original Transformer architecture. Each sub-layer is followed by a layer normalisation.[2]

Let us now describe the attention modules which independently form the Multi-Head Context Attention sub-layer.

3.1.1 Hierarchical Attention

Our hierachical attention module H-Attention(Q_s, Q_w, K_s, K_w, V_w) (Figure 1) is a reformulation of the Scaled Dot-Product Attention of Vaswani et al. (2017). Here, we have five inputs consisting of two types of keys and queries, one each for the sentences and the words, while the values are based only on words in the context. The Hierarchical Attention module has four operations:

1. **Sentence-level Key Matching:** This is performed on a set of queries simultaneously, packed together into a matrix Q_s. The sentence-level keys are also packed into a matrix K_s. We will describe in §3.3 how Q_s and K_s are computed. The attention weights are computed as:

$$\alpha_s = \text{sparsemax}(Q_s {K_s}^T/\sqrt{d_k}) \qquad (3)$$

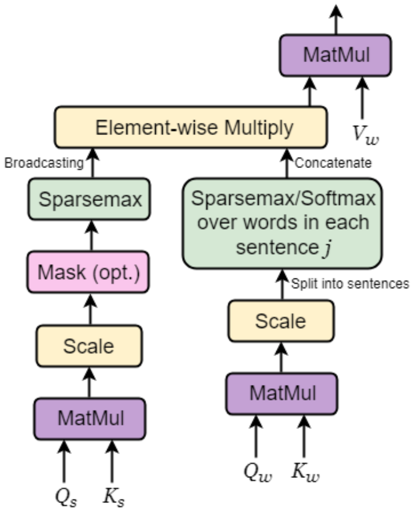

Figure 1: Hierarchical Context Attention module.

where d_k is the dimension of the keys, and α_s has dimensions equal to the total number of sentences in the document. We propose to use **sparsemax** (Martins and Astudillo, 2016), instead of softmax, as this gives us the intended selective attention behavior, that is identifying the key sentences that may potentially be relevant to the current sentence, hence making the model more efficient in compressing its memory. A softmax attention, on the other hand, can still assign low probability to sentences, forming a long-tail and absorbing significant probability mass, and it cannot fully *ignore* those sentences. An additive mask is used (before the *sparsemax* operation) based on whether we train for offline or online setting by masking out only the current sentence or current and future sentences, respectively.

2. **Word-level Key Matching:** Here the query and key matrices, Q_w and K_w, are word-level. We perform a word-level key matching for each sentence j in the document:

$$\alpha_w^j = \text{sparsemax}(Q_w {K_w^j}^T/\sqrt{d_k}) \qquad (4)$$

where α_w^j is the word-level attention vector for j^{th} sentence.[3] We can also use softmax, instead of sparsemax, for a coarser key matching. We explore the two variants in our experiments.

3. **Re-scaling attention weights:** The word-level attention is further re-weighted by the cor-

[2] We do not have residual connections after sub-layers in our Document-level Context Layer as we found them to have a deteriorating effect on the translation scores (also reported by Zhang et al. (2018)).

[3] This can be done for only the sentences with non-zero probabilities (obtained from the sentence-level key matching), however, we found it to be computationally expensive, as it required breaking down the batched matrices.

responding sentence-level attention (Nallapati et al., 2016) such that the probability of j^{th} sentence in a document is given by:

$$\alpha^{j}_{hier} = \alpha_s(j)\alpha^{j}_{w} \tag{5}$$

where $\alpha_s(j)$ is the attention weight for the j^{th} sentence obtained via Eq. 3 and α^{j}_{w} is as in Eq. 4. The re-weighting, thus, produces a scaled attention vector $\alpha_{hier} = \text{Concat}(\alpha^{1}_{hier}, ..., \alpha^{J}_{hier})$, each entry of which corresponds to the attention weight for a specific word in the document.

4. **Value Reading:** The set of word-level values is packed together into a matrix V_w and the matrix of outputs is given by $\alpha_{hier}V_w$. This multiplication, combined with sparsemax attention, allows to *prune* the hierarchy.

We further extend the MULTIHEAD attention function proposed by Vaswani et al. (2017) for our Hierarchical Attention module as:

$$\begin{aligned}\text{H-MULTIHEAD}(Q_s, K_s, Q_w, K_w, V_w) = \\ \text{Concat}(head_1, ..., head_H)W^O\end{aligned}$$

where $head_h = \text{H-Attention}(Q_sW^{Q_s}_h, Q_wW^{Q_w}_h, K_sW^{K_s}_h, K_wW^{K_w}_h, V_wW^{V_w}_h)$, W's are parameter matrices and all (five) inputs are transformed using separate linear layers.

3.1.2 Flat Attention

Another way to model the context $\boldsymbol{D}_{-j}$ is via single-level attention by re-using the Scaled Dot-Product Attention in Vaswani et al. (2017),

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T/\sqrt{d_k})V \tag{6}$$

The attention[4] here is of two types: (i) *sentence-level* if K, V are computed for sentences in the document, or (ii) *word-level* if K, V are computed for words in the document. The former module is similar to the Memory Networks architecture of Maruf and Haffari (2018) in that it uses sentence-level information. However, there are two key differences: (i) we use MultiHead attention as in the Transformer architecture, and (ii) our context attention is dynamic such that we have a separate attention for each query word.

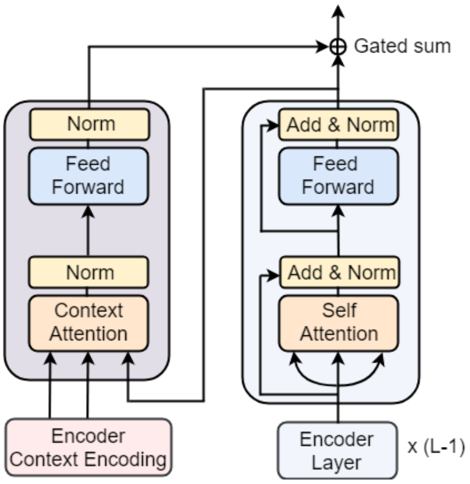

Figure 2: Encoder-side context integration.

3.2 Context Gating

As mentioned previously, the Multi-Head Context Attention sub-layer is part of the Context Layer (Figure 2), the output of which is fed into the Transformer architecture through context gating (Tu et al., 2018). For i^{th} word in source or target:

$$\gamma_i = \sigma(W_r r_i + W_d d_i) \tag{7}$$

$$\tilde{r}_i = \gamma_i \odot r_i + (1 - \gamma_i) \odot d_i \tag{8}$$

where W's are parameter matrices, r_i is the output of encoder or decoder stack for i^{th} word, d_i is the output from the context layer for i^{th} word and $\tilde{r}_i$ is the final hidden representation for the same.

3.3 Integrated Model

The context can be integrated into the encoder or decoder of the NMT model depending on if it is monolingual or bilingual.[5]

Monolingual context integration in Encoder We add the Document-level Context Layer alongside the encoder stack as shown in Figure 2. The Encoder Context Encoding block stores the keys and values produced from the pre-trained sentence-level NMT model. For word-level attention, the keys K_w and values V_w are composed of vector representations (from last encoder layer) of source words in the document, while for the sentence-level attention, the keys K_s and values V_s are composed of vector representations of sentences in the document where the vector representation of each sentence is an average of the word

[4] We plan to investigate sparse flat attention in future work.

[5] We do not integrate context into both encoder and decoder as it would have redundant information from the source (the context incorporated in the decoder is bilingual), in addition to increasing the complexity of the model.

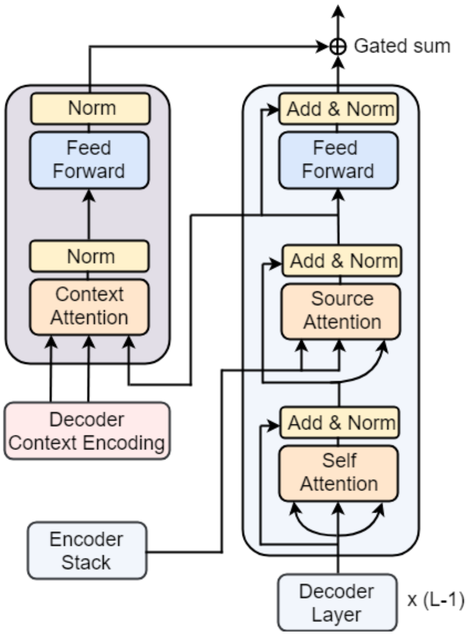

Figure 3: Decoder-side context integration.

representations in that sentence. The queries Q_w, Q_s are linear transformations of the output of the L^{th} encoder layer which are then matched with the corresponding keys and values stored in the Encoder Context Encoding block just described.

Bilingual context integration in Decoder We again add the Document-level Context Layer alongside the decoder stack as in Figure 3. However, instead of choosing the keys and values to be monolingual as in the encoder, we follow Tu et al. (2018) in choosing the key to match to the source-side context, while designing the value to match to the target-side context. Hence, the keys (in the Decoder Context Encoding block) are composed of context vectors from the Source Attention sub-layer, while the values are composed of the hidden representations of the target words, both from the last decoder layer. Again the keys K_w and K_s are either for individual target words or target sentences, and same goes for V_w and V_s. The queries Q_w, Q_s for the Context Layer come from the Source Attention sub-layer in the L^{th} layer of the decoder (Figure 3).

4 Experiments

4.1 Setup

Datasets We conduct experiments for English→German on three different domains: TED talks, News-Commentary and Europarl. These datasets are chosen based on their variance in genre, style and level of formality:

Domain	#Sentences	Document length
TED	0.21M/9K/2.3K	120.89/96.42/98.74
News	0.24M/2K/3K	38.93/26.78/19.35
Europarl	1.67M/3.6K/5.1K	14.14/14.95/14.06

Table 1: Training/development/test corpora statistics: number of sentences (K stands for thousands and M for millions), and average document length (in sentences).

- **TED** This corpus is from the IWSLT 2017 MT track (Cettolo et al., 2012) and contains transcripts of TED talks aligned at sentence level. Each talk is considered to be a document. We combine *tst2016-2017* into the test set and the rest are used for development.
- **News-Commentary** We obtain the sentence-aligned document-delimited News Commentary v11 corpus for training.[6] The WMT'16 *news-test2015* and *news-test2016* are used for development and testing, respectively.
- **Europarl** This dataset is extracted from Europarl v7 (Koehn, 2005). The source and target sentences are aligned using the links provided by Tiedemann (2012). Following Maruf and Haffari (2018), we use the *SPEAKER* tag as the document delimiter. Documents longer than 5 sentences are kept and the resulting corpus is randomly split into training, dev and test sets.

The corpora statistics are provided in Table 1. All datasets are tokenised and truecased using the Moses toolkit (Koehn et al., 2007), and split into subword units using a joint BPE model with 30K merge operations (Sennrich et al., 2016).

Models and Baselines For offline document MT, we have two context-agnostic baselines: (i) a modified version of RNNSearch (Bahdanau et al., 2015), which incorporates dropout on the output layer and improves the attention model by feeding the previously generated word, and (ii) the state-of-the-art Transformer architecture. For the online case, we again have the Transformer as a context-agnostic baseline and two context-aware baselines (Zhang et al., 2018; Miculicich et al., 2018).

All models are implemented in C++ using DyNet (Neubig et al., 2017). For RNNSearch, we modify the sentence-based NMT implementation in *mantis* (Cohn et al., 2016). The encoder is a single layer bidirectional GRU (Cho et al., 2014) and

[6] www.casmacat.eu/corpus/news-commentary.html

	Integration into Encoder						Integration into Decoder					
	TED		News		Europarl		TED		News		Europarl	
Model	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**
RNNSearch	19.24	40.81	16.51	36.79	26.26	44.14	19.24	40.81	16.51	36.79	26.26	44.14
Transformer	23.28	44.17	22.78	42.19	28.72	46.22	23.28	44.17	22.78	42.19	28.72	46.22
+Attention, sentence	24.47	**45.25**	**24.78**	43.90	29.60	46.98	**24.38**	44.82	24.67	43.82	29.67	**47.04**
word	**24.55**	44.89	24.55	43.75	29.63	46.94	24.27	44.95	24.23	43.44	29.68	46.93
+H-Attention, sparse-soft	24.23	44.81	24.76	44.10	**29.72**	47.03	24.19	44.94	**24.67**	**43.86**	**29.69**	46.97
sparse-sparse	24.27	45.07	24.66	**44.18**	29.64	**47.04**	24.14	**45.32**	24.49	43.49	29.59	47.02

Table 2: BLEU and Meteor scores for variants of our model and two context-agnostic baselines for offline document MT. **bold**: Best performance. All reported results for our model are significantly better than both baselines.

	Integration into Encoder						Integration into Decoder					
	TED		News		Europarl		TED		News		Europarl	
Model	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**	**BLEU**	**Meteor**
Zhang et al. (2018)	24.00	44.69	23.08	42.40	29.32	46.72	23.82	44.54	22.78	42.17	29.35	46.73
Miculicich et al. (2018)	**24.58**	**45.48**	**25.03**	44.02	28.60	46.09	24.39	45.23	24.38	43.58	29.58	46.91
Transformer	23.28	44.17	22.78	42.19	28.72	46.22	23.28	44.17	22.78	42.19	28.72	46.22
+Attention, sentence	24.38	45.01	24.46*	43.46*	29.59$^\clubsuit$	47.02$^\clubsuit$	24.29*	45.13*	**24.75**$^\clubsuit$	**44.03**$^\clubsuit$	29.56	46.84
word	24.22	45.05*	24.84*	**44.27***	29.67$^\clubsuit$	47.04$^\clubsuit$	24.02	44.79	24.17*	43.53*	**29.90**$^\clubsuit$	47.11$^\clubsuit$
+H-Attention, sparse-soft	24.34	45.05*	24.54*	43.66*	**29.75**$^\clubsuit$	**47.22**$^\clubsuit$	**24.62***	**45.32***	24.36*	43.67*	29.80*	**47.11**$^\clubsuit$
sparse-sparse	24.42	45.38*	24.73*	44.06*	29.39$^\diamondsuit$	46.78$^\diamondsuit$	24.43*	45.10*	24.58*	43.75*	29.64*	46.94*

Table 3: BLEU and Meteor scores for variants of our model and three baselines for online document MT. **bold**: Best performance. $\star$, $\diamondsuit$, $\clubsuit$: Statistically significantly better than our implementations of Zhang et al. (2018), Miculicich et al. (2018), or both. All reported results for our model are significantly better than the Transformer.

the decoder is a 2-layer GRU with embeddings and hidden dimensions set to 512. The dropout rate for the output layer is set to 0.2. For the Transformer, we use *Transformer-DyNet*[7] implementation and extend it for our context-aware NMT model.[8] The hidden dimensions and feed-forward layer size is set to 512 and 2048 respectively. We use 4 layers[9] each in the encoder and decoder with 8 attention heads and employ label smoothing with a value of 0.1. We also employ all four types of dropouts as in the original Transformer with a rate of 0.1 for the sentence-based model and 0.2 for our context-aware model.

For training all models, we use the default Adam optimiser (Kingma and Ba, 2015) with an initial learning rate of 0.0001 and employ early stopping. For our context-aware NMT model, we use a two-stage training strategy as described in §2.2. For inference, we use Iterative Decoding only when using the bilingual context. All experiments are run on a single Nvidia P100 GPU with 16GBs of memory.[10]

Evaluation Metrics For evaluation, we use BLEU (Papineni et al., 2002) and Meteor (Lavie and Agarwal, 2007) scores on tokenised text, and measure statistical significance with respect to the baselines, $p < 0.05$ (Clark et al., 2011).

4.2 Main Results

We divide our experiments into two parts: offline and online document MT.

Offline Document MT From the scores of the two context-agnostic baselines in Table 2, we can see that the Transformer beats the RNNSearch model in all cases by atleast +2.5 BLEU and +2.1 Meteor scores showing that our hyperparameter choice for the Transformer is indeed effective.

For the Encoder Context integration, our Hierarchical Attention models perform the (near) best for News and Europarl datasets with +1.98 and +1 BLEU and +1.99 and +0.82 Meteor improvements with respect to the Transformer. For TED talks, however, we find the Flat Attention based models (sentence and word-level) to be the best with +1.27 BLEU and +1.08 METEOR improvements. For Decoder Context integration, we find the Hierarchical Attention to be the best in majority of the cases both in terms of BLEU and Meteor.

[7] https://github.com/duyvuleo/Transformer-DyNet

[8] The code is available at https://github.com/sameenmaruf/selective-attn

[9] We found this configuration to be much more stable than using 6 layers with almost no difference in performance as reported by Xia et al. (2018).

[10] The experiments can also be run on GPUs with 10-12GBs of memory by reducing the batch size at the expense of increased computational cost.

Online Document MT From Table 3, all our models significantly outperform the context-agnostic baseline and are significantly better than Zhang et al. (2018) in majority cases. For Encoder Context integration, the HAN encoder (Miculicich et al., 2018) is the best for TED and News datasets, however, the results are statistically insignificant with respect to our best model. For Europarl, our Hierarchical Attention model performs significantly better than Miculicich et al. (2018) with a gain of +1.15 BLEU and +1.13 Meteor. For Decoder Context integration, our Hierachical Attention models are the winner in majority cases and our best models beat Miculicich et al. (2018) for all datasets based upon BLEU and Meteor. The main conclusion we draw from these results is that efficiently using the context information at hand is crucial when it comes to improving the performance of context-aware NMT. Furthermore, shorter pieces of text (e.g., the ones in Europarl) benefit more from using global context because their sentences may exhibit higher interdependency than those in a longer piece of text.

Offline vs. Online Document MT Let us compare the overall results for the offline and online document MT settings. For all datasets and model variants, we find the best BLEU and Meteor scores in Tables 2 and 3 (highlighted in bold) to be quite close to each other with those for the online setting slightly better. This is quite self-explanatory, because in essence, all of the datasets comprise of talks, speeches or commentaries, which are in fact produced in an online manner and hence we do not see drastic improvements in terms of BLEU and Meteor when conditioning on the future context. This, in our opinion, does not mean that we should never look into the future, but just that NMT models in general are highly subjective to data, and whether context-aware models benefit from future context is also dependent on that.

4.3 Analysis

Evaluation on Contrastive Pronoun Test Set It has been argued that evaluation metrics which quantify the overall translation quality are somewhat ill-equipped to assess how well models translate inter-sentential phenomena such as pronouns. Hence, we use a test suite of contrastive translations designed to measure accuracy of translating the English pronoun *it* to its German counterparts *es*, *er* and *sie* (Müller et al., 2018). We are interested to see if our global document-context models surpass the local context-aware baselines. Table 4 shows that not only our global-context models are quite effective but our Hierarchical Attention model is most useful when the antecedent is farther than three previous sentences. We also conclude that models for offline MT perform better when antecedent distance is greater than two.

Model	**antecedent distance**				
	0	1	2	3	>3
	Offline document MT				
RNNSearch	0.415	0.310	0.424	0.440	0.647
Transformer	0.586	0.308	0.437	0.48	0.642
+Attention, sentence	0.677	0.314	0.439	0.478	0.697
word	**0.686**	**0.347**	**0.464**	**0.511**	0.679
+H-Attention, sparse-soft	0.676	0.308	0.440	0.480	0.686
sparse-sparse	0.652	0.303	0.435	0.471	**0.701**
	Online document MT				
Zhang et al. (2018)	0.622	0.321	0.450	0.485	0.658
Miculicich et al. (2018)	0.722	0.326	0.451	0.471	0.661
Transformer	0.586	0.308	0.437	0.48	0.642
+Attention, sentence	**0.732**	**0.340**	**0.460**	0.485	0.661
word	0.690	0.317	0.444	0.487	0.683
+H-Attention, sparse-soft	0.692	0.329	0.446	0.464	0.656
sparse-sparse	0.711	0.317	0.437	**0.489**	**0.692**

Table 4: Accuracy on contrastive test set with regard to antecedent distance (in sentences) on TED Talks. Antecedent distance 0 means the pronoun occurs in the same sentence as the antecedent.

Subjective Evaluation We conduct a subjective evaluation to validate the benefit of exploiting document-level context. Three native German speakers were asked to choose the better (with ties allowed) of two translations for each of 18 documents (randomly sampled from Europarl test set). The two translations, one produced by the Transformer and the other by our Hierarchical Attention model, were evaluated in terms of: *adequacy* (Which translation expresses the meaning of the source text more adequately?) and *fluency* (Which text has better German?) (Läubli et al., 2018). Let a, b be number of ratings in favour of Transformer or our model, respectively, and t be number of ties, then number of successes $x = b + 0.5t$ and trials $n = a + b + t$. We test for statistically significant preference of our model over the Transformer by means of two-sided Sign Tests and find that our model is better than the Transformer both in terms of document-level adequacy (x = 39, n = 54, p = 0.0015) and fluency (x = 38, n = 54, p = 0.0038).

Model Complexity Model complexity is reported in Table 5. Our context-aware models introduce only 8% more parameters to the original

Model	#Params	Speed (words/sec.)	
		Training	Decoding
Zhang et al. (2018)	59.5M	3300	84.94
Miculicich et al. (2018)	54.8M	1650	76.90
Transformer	50M	5100	86.33
+Attention, sentence	53.7M	3750	83.84
+H-Attention, sparse-soft	54.2M	2600	74.11

Table 5: Model complexity for Encoder Context integration models (News-Commentary).

Transformer model. In comparison to the Transformer, our Hierarchical Attention model is slow in training, dropping the speed by almost 50%[11], but it is still almost 40% faster than Miculicich et al. (2018). At decoding time, our Hierarchical Attention model is almost equivalent to Miculicich et al. (2018) and only 13% slower than Zhang et al. (2018). Hence, attending to the whole document (instead of few previous sentences) does not add to the time complexity of the model on average.

Qualitative Analysis To analyse the effect of using sparse attention at both the sentence and word-level, we looked at the attention weights computed by *sparsemax*. Table 6 shows an example where our model helped generate a correct translation of the noun "thoughts" (highlighted in bold). The context sentences shown in the bottom box had the highest attention weights as assigned by sparsemax. It seems that this particular attention head focuses more on phrases like "words of sympathy", "support', "symbol of hope" which are related to the query "thoughts". Another example in Table 7 shows how our model correctly translates the pronoun "their". Upon looking at the words in the context sentences, it seems that this particular attention head focuses on the words related to the antecedent "Croatia's Serbian population" with most of the weight concentrated around neighbouring words in sentence s^{j-1}. It is evident from both examples that word-level sparsity is more prevalent in longer sentences in the context. The same holds for sparsity at sentence-level.

5 Related Work

The body of work in document-level MT can be broadly classified into two categories: conventional MT and neural MT.

Src: my **thoughts** are also with the victims . Ref: meine **Gedanken** sind auch bei den Opfern .
Transformer: ich **denke** auch an die Opfer . Zhang et al. (2018): ich **denke** auch an die Opfer . Miculicich et al. (2018): ich **denke** auch an die Opfer . Our Model: meine **Gedanken** sind auch bei den Opfern .

Head 2: Attention to related words *sympathy, support, hope*
s^{j-2}: (FR) Madam President , many things have already been said , but I would like to echo all the words of sympathy and support that have already been addressed to the peoples of Tunisia and Egypt .
s^{j+4}: it must implement a strong strategy towards these countries .
s^{j-1}: they are a symbol of hope for all those who defend freedom .

Table 6: Example of noun disambiguation. Source context sentences are ordered in decreasing probability mass. The intensity of color corresponds to the attention given to a specific word before rescaling.

Src: Croatia is **their** homeland , too . Ref: Kroatien ist auch **ihre** Heimat .
Transformer: Kroatien ist auch **seine** Heimat . Our Model: Kroatien ist auch **ihr** Heimatland .

Head 8: Attention to words related to the antecedent.
s^{j-1}: to name but a few , these include cooperation with the Hague Tribunal , efforts made so far in prosecuting corruption , restructuring the economy and finances and greater commitment and sincerity in eliminating the obstacles to the return of Croatia 's Serbian population .
s^{j-4}: by signing a border arbitration agreement with its neighbour Slovenia , the new Croatian Government has not only eliminated an obstacle to the negotiating process , but has also paved the way for the resolution of other issues .

Table 7: Example of pronoun disambiguation. Context sentences are ordered in decreasing probability mass.

Conventional Document-level MT These can further be classified into two main categories. The first, which use cache-based memories (Tiedemann, 2010; Gong et al., 2011) and the second, which focus on specific discourse phenomema like anaphora (Hardmeier and Federico, 2010), lexical cohesion (Xiong et al., 2013; Gong et al., 2015; Mascarell, 2017) and coreference (Miculicich Werlen and Popescu-Belis, 2017) to name a few. Most of these approaches are, however, restrictive as they mostly involve using handcrafted features similar to the conventional MT approaches.

Document-level Neural MT The works here can again be divided into two categories: *online*—use previous context only, and *offline*—use both past and future contexts. Most works fall into the former category, with those that use only a single

[11] DyNet implementation of *sparsemax* is CPU-based and only operates on column vectors. We believe a GPU-based matrix implementation would bring the speed much closer to our Word Attention model (training: 3100, decoding: 81.38).

previous sentence in the source (Jean et al., 2017; Tiedemann and Scherrer, 2017; Voita et al., 2018); one previous sentence both in source and target (Bawden et al., 2018); more than one previous source sentence (Wang et al., 2017; Zhang et al., 2018); or a few previous source and target sentences (Miculicich et al., 2018). Apart from fixing the context length, there are few works which use cache-based memories to store contextual information (Tu et al., 2018; Kuang et al., 2018) and use that to improve the MT system performance. A recent work (Maruf et al., 2018) reports promising results when using the complete history for translating online conversations.

For the offline setting, however, there is only one work that effectively uses the full document-context on both source and target-side using memory networks (Maruf and Haffari, 2018). The debate in document-level NMT today is mostly about how much of the previous context to use and there has been no comparison between the online and offline setting except using only one previous and following sentence (Voita et al., 2018).

Sparse Attention Sparse attention and its constrained variants have been used to address the coverage problem in NMT (Malaviya et al., 2018) by limiting the amount of attention that each source word can receive. Apart from NMT, sparse attention has been shown to yield promising results for NLP tasks of textual entailment (Martins and Astudillo, 2016) and summarization (Niculae and Blondel, 2017).

6 Conclusion

We have proposed a novel approach to hierarchical attention for context-aware NMT, based on sparse attention, which is both scalable and efficient. Experiments and evaluation on three English→German datasets in offline and online document MT settings show that our approach surpasses context-agnostic and two recent context-aware baselines. The qualitative analysis shows that the sparsity at sentence-level allows our model to identify key sentences in the document context and the sparsity at word-level allows it to focus on key words in those sentences allowing for an efficient compression of memory. In future work, we plan to dig deeper on the benefits of sparse attention in terms of better interpretability of context-aware NMT models.

Acknowledgments

The authors are grateful to the anonymous reviewers for their helpful comments and corrections. SM would like to thank her colleagues at Monash University: Veronika Kuchta, Harald Bögeholz and Hagen Lauer, for their help in the subjective evaluation. This work was supported by the Multi-modal Australian ScienceS Imaging and Visualisation Environment (MASSIVE) (www.massive.org.au), the European Research Council (ERC StG DeepSPIN 758969), the Fundação para a Ciência e Tecnologia through contracts UID/EEA/50008/2019 and CMUPERI/TIC/0046/2014 (GoLocal), and a Google Faculty Research Award to GH.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*.

Rachel Bawden, Rico Sennrich, Alexandra Birch, and Barry Haddow. 2018. Evaluating discourse phenomena in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1304–1313. Association for Computational Linguistics.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy.

Kyunghyun Cho, B van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (Short Papers)*, pages 176–181. Association for Computational Linguistics.

Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. 2016. Incorporating structural alignment biases into an attentional neural translation model. In *Proceedings of the North American Chapter of* *the Association for Computational Linguistics: Human Language Technologies*, pages 876–885. Association for Computational Linguistics.

Peter Dayan, Sham Kakade, and P. Read Montague. 2000. Learning and selective attention. *Nature Neuroscience*, 3:1218–1223.

Zhengxian Gong, Min Zhang, and Guodong Zhou. 2011. Cache-based document-level statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 909–919. Association for Computational Linguistics.

Zhengxian Gong, Min Zhang, and Guodong Zhou. 2015. Document-level machine translation evaluation with gist consistency and text cohesion. In *Proceedings of the Second Workshop on Discourse in Machine Translation*, pages 33–40, Lisbon, Portugal. Association for Computational Linguistics.

Christian Hardmeier and Marcello Federico. 2010. Modelling pronominal anaphora in statistical machine translation. In *International Workshop on Spoken Language Translation*, pages 283–289.

Sebastien Jean, Stanislas Lauly, Orhan Firat, and Kyunghyun Cho. 2017. Does neural machine translation benefit from larger context? In *arXiv:1704.05135*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Conference Proceedings: the 10th Machine Translation Summit*, pages 79–86. AAMT.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. Association for Computational Linguistics.

Shaohui Kuang, Deyi Xiong, Weihua Luo, and Guodong Zhou. 2018. Modeling coherence for neural machine translation with dynamic and topic caches. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 596–606. Association for Computational Linguistics.

Samuel Läubli, Rico Sennrich, and Martin Volk. 2018. Has machine translation achieved human parity? A case for document-level evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 4791–4796. Association for Computational Linguistics.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231. Association for Computational Linguistics.

Chaitanya Malaviya, Pedro Ferreira, and André F. T. Martins. 2018. Sparse and constrained attention for neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 370–376.

André F. T. Martins and Ramón Astudillo. 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48, pages 1614–1623, New York, New York, USA. PMLR.

Sameen Maruf and Gholamreza Haffari. 2018. Document context neural machine translation with memory networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1275–1284. Association for Computational Linguistics.

Sameen Maruf, André F. T. Martins, and Gholamreza Haffari. 2018. Contextual neural model for translating bilingual multi-speaker conversations. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 101–112, Brussels, Belgium. Association for Computational Linguistics.

Laura Mascarell. 2017. Lexical chains meet word embeddings in document-level statistical machine translation. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 99–109. Association for Computational Linguistics.

Lesly Miculicich, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2947–2954.

Lesly Miculicich Werlen and Andrei Popescu-Belis. 2017. Using coreference links to improve spanish-to-english machine translation. In *Proceedings of the 2nd Workshop on Coreference Resolution Beyond OntoNotes (CORBON 2017)*, pages 30–40, Valencia, Spain. Association for Computational Linguistics.

Mathias Müller, Annette Rios, Elena Voita, and Rico Sennrich. 2018. A large-scale test set for the evaluation of context-aware pronoun translation in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 61–72, Belgium, Brussels. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of Conference on Natural Language Learning*, pages 280–290. Association for Computational Linguistics.

Graham Neubig, Chris Dyer, Yoav Goldberg, Austin Matthews, Waleed Ammar, Antonios Anastasopoulos, Miguel Ballesteros, David Chiang, Daniel Clothiaux, Trevor Cohn, Kevin Duh, Manaal Faruqui, Cynthia Gan, Dan Garrette, Yangfeng Ji, Lingpeng Kong, Adhiguna Kuncoro, Gaurav Kumar, Chaitanya Malaviya, Paul Michel, Yusuke Oda, Matthew Richardson, Naomi Saphra, Swabha Swayamdipta, and Pengcheng Yin. 2017. Dynet: The dynamic neural network toolkit. *arXiv preprint arXiv:1701.03980*.

Vlad Niculae and Mathieu Blondel. 2017. A regularized framework for sparse and structured neural attention. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3338–3348. Curran Associates, Inc.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pages 3104–3112. MIT Press.

Jörg Tiedemann. 2010. Context adaptation in statistical machine translation using models with exponentially decaying cache. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, DANLP 2010, pages 8–15, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Jörg Tiedemann and Yves Scherrer. 2017. Neural machine translation with extended context. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 82–92. Association for Computational Linguistics.

Zhaopeng Tu, Yang Liu, Shuming Shi, and Tong Zhang. 2018. Learning to remember translation history with a continuous cache. *Transactions of the Association for Computational Linguistics*, 6:407–420.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. 2018. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1264–1274. Association for Computational Linguistics.

Longyue Wang, Zhaopeng Tu, Andy Way, and Qun Liu. 2017. Exploiting cross-sentence context for neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 2816–2821. Association for Computational Linguistics.

Yingce Xia, Xu Tan, Fei Tian, Tao Qin, Nenghai Yu, and Tie-Yan Liu. 2018. Model-level dual learning. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5379–5388.

Deyi Xiong, Yang Ding, Min Zhang, and Chew Lim Tan. 2013. Lexical chain based cohesion models for document-level statistical machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1563–1573. Association for Computational Linguistics.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489. Association for Computational Linguistics.

Jiacheng Zhang, Huanbo Luan, Maosong Sun, Feifei Zhai, Jingfang Xu, Min Zhang, and Yang Liu. 2018. Improving the transformer translation model with document-level context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 533–542. Association for Computational Linguistics.